29/3/2020

Jonathon Samson-Noble (11000606)

Otago Polytechnic

IN723 Advance Networks

Research Assignment – Network Automation



**Contents**

[**Introduction** 3](#_Toc36499869)

[**What is Network Automation** 3](#_Toc36499870)

[**What is Ansible?** 3](#_Toc36499871)

[**Why use Ansible?** 4](#_Toc36499872)

[**How to use Ansible?** 5](#_Toc36499873)

[**Requirements** 5](#_Toc36499874)

[**Inventory** 5](#_Toc36499875)

[**Playbooks** 6](#_Toc36499876)

[**Integrations** 7](#_Toc36499877)

[**Versions** 7](#_Toc36499878)

[**Ansible** 7](#_Toc36499879)

[**Ansible Tower** 8](#_Toc36499880)

[**Ansible AWX** 8](#_Toc36499881)

[**Advantages** 8](#_Toc36499882)

[**Disadvantages** 8](#_Toc36499883)

[**Conclusion** 9](#_Toc36499884)

[**References** 10](#_Toc36499885)

# **Introduction**

Modern IT networks are scaling rapidly and require a business to react swiftly and accordingly to stay relevant this can be achieved through the use of network automation tools such as Ansible, allowing for businesses to deploy updates or new services quickly across their network.

The following paper will be on network automation with a primary focus on the Ansible software tool, we will discuss what is network automation, what is ansible, why you might use ansible, how to use ansible and some of the key components of it, the different versions of the ansible software and the advantages and disadvantages.

# **What is Network Automation**

Network automation is the process of automating the configuration, management, testing, deployment, and operations of physical and virtual devices within a network. Everyday network tasks and functions are performed automatically. Using a combination of hardware and software-based solutions, large organizations, service providers, and enterprises can implement network automation to control and manage repetitive processes and improve network service availability.[[1]](https://www.juniper.net/us/en/products-services/what-is/network-automation/)

# **What is Ansible?**

Ansible is an open source agentless automation tool that can be used for such tasks as configuration management, application deployment and provisioning.

Ansible works by creating an inventory of hosts which it then uses in a playbook to deploy pre-built modules or user created modules against those hosts it achieves this by temporarily connecting via ssh or windows remote management to your targeted hosts, an example of this can be seen below in figure 1.

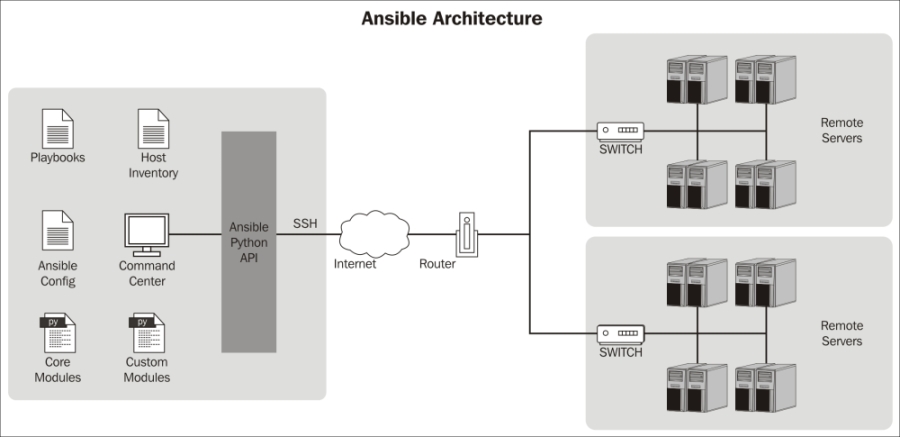


Figure 1 Ansible Diagram [[2]](https://subscription.packtpub.com/book/networking_and_servers/9781783550630/1/ch01lvl1sec09/the-ansible-architecture)

# **Why use Ansible?**

Ansible is a free open-source agentless tool that will naturally reduce the resources consumed and overhead of the network devices used because, Ansible will only need to be installed on the control nodes and no additional software is needed on the managed nodes.

Ansible is simple to learn as the playbooks required to run the configuration tasks are both easy to use and readable, they are based on YAML (YAML Ain’t Markup Language) format, a descriptive language.

As your business grows over time staff will come and go and you may not have the original network engineers and you will have most likely added either new sites or network equipment to your business which will have been configured differently from pre-existing equipment, this can lead to security problems and out of date versions of software or even deprecated software on your devices.

Ansible innately allows for consistent and standardized environments across your network as you can place your nodes into specific groups, for example, you could have a database group and a web group which would all be configured based on your playbook configuration. This in turn will lead to a more agile environment and a reduction in human error because there is less inherent risk due to the administrators of the network having to code less because every line of code is a potential bug.

It also allows less experienced network administrators on your team to run complex playbooks or scripts as they would have already been thoroughly tested beforehand.

By automating everyday network tasks with Ansible this will allow your network engineers more time to focus on more crucial tasks which will turn increase the productivity of your business.

Ansible can be integrated into a wide range of existing vendor technologies that already may be required on your network.

It is also used by such companies as Lockheed Martin[[3]](https://www.ansible.com/network-automation-of-f5-big-ip-devices-with-ansible-tower), Apple[[4]](https://www.ansible.com/blog/enterprise-ansible) and NASA.[[5]](https://www.ansible.com/blog/nasa-automation)

# **How to use Ansible?**

Ansible is very simple to setup as it only requires that the control nodes have the Ansible software installed on it.

## **Requirements**

In order to use Ansible for your network automation solution your control node must have the following requirements met.

* Linux/Unix based system[[10]](https://docs.ansible.com/ansible/latest/installation_guide/intro_installation.html#prerequisites)
* Python 2 (version 2.7) or Python 3 (version 3.5 or higher) installed[[10]](https://docs.ansible.com/ansible/latest/installation_guide/intro_installation.html#prerequisites)

Microsoft windows is currently not supported for the control node.

Documentation for installing Ansible on your given Linux/Unix system can be found at <https://docs.ansible.com/ansible/latest/installation_guide/intro_installation.html>

**Inventory**

In Ansible the inventory is a description of all the nodes that can be accessed by Ansible, this file can either be in YAML or INI format. The inventory configuration file can have either the hostname or the ip address of each node. Nodes inside the inventory file can also be assigned into a group or multiple groups which are designated by the group name “[*group name*]”

An example of the inventory file which is stored in /etc/ansible/hosts can be seen in figure 2 below.

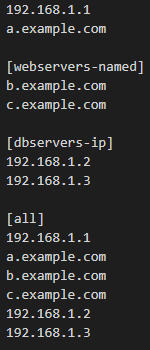
The best idea when creating your inventory file is to have a standardized logical naming convention for your groups this will allow you in the future to add additional devices to that group if they qualify for that group, for example, if you needed to add an extra database server down the line you can add that hostname or ip to your database group in your inventory file.

Figure 2 Example of an inventory file

By assigning the nodes to groups this allows you to utilize Ansible playbooks against a group rather than an individual node this will allow for consistent configuration across all of your machines in a given group and increase productivity as you’re connecting to machines individually to configure them one by one , for example, if you want to make sure all of your nodes in your [dbservers-ip] group were running the same configuration. This also has the added benefit of standardizing your network

## **Playbooks**

Ansible playbooks can be used to manage the configurations of and deployments to remote machines. At a more advanced level, they can sequence multi-tier rollouts involving rolling updates, and can delegate actions to other hosts, interacting with monitoring servers and load balancers along the way.[[6]](https://docs.ansible.com/ansible/latest/user_guide/playbooks.html)

Creating playbooks for each individual task rather than having a playbook containing multiple tasks will allow for an improved the readability for users, but if you need to run multiple tasks you can import playbooks into other playbooks allowing you to run multiple tasks.

By creating individual task playbooks it has the added advantage of being able to create a customized playbook rapidly using playbooks you have already created, for example, if you needed to create a new database server that also has some form of web application you can take your database task playbooks and combine them with your web task playbooks importing them into a new custom playbook that you know will work because your reusing previous playbooks have already been tested.

You can also restrict it so that only specified users can run certain playbooks. The way to do this is by assigning roles and permissions, This will help to improve the security of your system as you can limit the more critical and potential system breaking playbooks to authorized users.

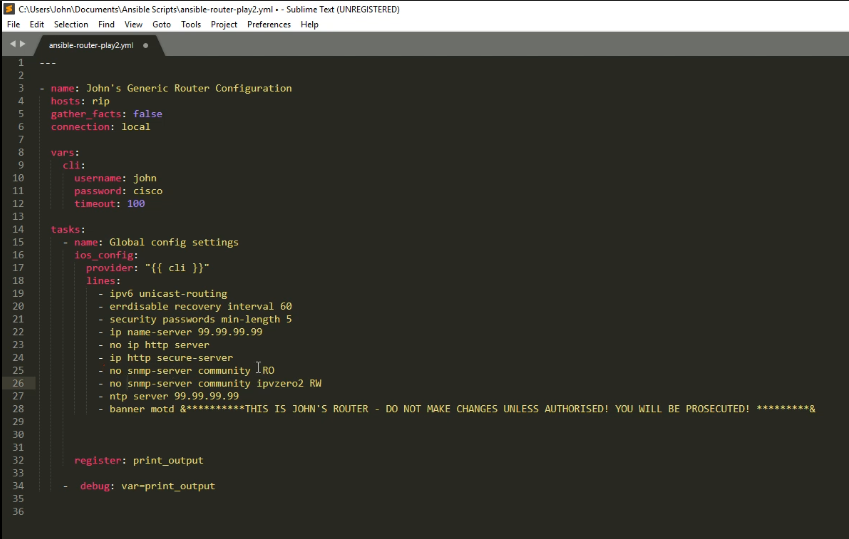


Figure 3 Playbook example of a cisco router configuration [[7]](https://www.youtube.com/watch?v=xw4IinFGVzA)

The Ansible documentation also contains information on best practices when creating your own playbooks which can be found here <https://docs.ansible.com/ansible/latest/user_guide/playbooks_best_practices.html>

## **Integrations**

Ansible can be integrated into existing infrastructure, networks , containers , cloud and DevOps tools for which it includes hundreds of modules to support a wide range of vendors and their technologies such as cisco, AWS, Microsoft Azure, VMware to name a few, a more in depth list can be found here at <https://www.ansible.com/integrations>

# **Versions**

There are a few different versions of the Ansible software ranging from both free and paid versions and CLI to GUI.

**Ansible** – Open-source command line only version of Ansible that is free, it contains all the functionality minus the web-based GUI that Ansible tower offers

**Ansible Tower** – free for up to 10 nodes features web-based GUI but requires that Red Hat Enterprise Linux 7 or 8 be installed on the device running Ansible Tower more information relating to ansible tower can be found here <https://www.ansible.com/products/tower>

**Ansible AWX** – This is the open source version of the Ansible Tower web application which is free, more information relating can be found here <https://github.com/ansible/awx>

# **Advantages**

Some of the main advantages to using Ansible for network automation are the following:

* Ansible is a free open-source automation tool that can be integrated with a wide range of existing vendor tools, they have extensive documentation on modules that can be used with these tools.
* Due to Ansible being an agentless tool this means there is less overhead, and resources being used as only the control nodes require Ansible to be installed.
* Multiple control nodes allow for redundancy so if your main Ansible server was disconnected you would still be able to function on your backup control nodes, by also having multiple control nodes in different parts of your network you could have a script that is executed when you run your playbook on your main device that would contact another Ansible control node and deploy that same playbook to a given segment thus reducing the load on each control node.
* Automation helps to reduce human error by requiring less coding from users.
* Reduction in outages due to Ansible innately creating a standardized network which allows for improved troubleshooting times.
* If you need to update software on your network devices or deploy new services for your clients, this can easily be done by creating new playbooks and deploying it to your network allowing the business to respond rapidly to required changes.
* Scalability, this can easily be achieved, because if your adding more infrastructure to your network you already have the playbooks created from your existing devices and you just need to add the new devices to your inventory file.

# **Disadvantages**

Some of the main disadvantages to network automation would be the following

* The initial cost of learning to use a network automation tool like Ansible as well as the cost of implementing a network automation solution, like any IT solution will require development time, the business will not see the full benefits until deployment.
* Due to the nature of how Ansible works there may be network performance issues when deploying a playbook to your whole network, this can easily be dealt with by having multiple control nodes in different segments of your network thanks to Ansible allowing multiple control nodes in your network.
* By relying on a network automation tool to maintain your network management and deployments among other tasks, you run the risk that if the tool was to go down you may not have solutions in place to maintain your network if some time has passed since its implementation.

# **Conclusion**

As modern IT environments continue to become more complex by implementing a range of physical and cloud-based technologies and scaling larger in size, network administrators and businesses will need to start preparing to implement network automation solutions that will allow them to maintain a consistent and standardized network environment whilst also having the ability to scale their network.

Whilst Ansible can do all this for your business it will not be an overnight fix and will require both time and investment in order to develop and deploy this solution.

# **References**

1. What is Network Automation. (n.d.). Retrieved from <https://www.juniper.net/us/en/products-services/what-is/network-automation/>
2. The Ansible architecture (n.d.). Retrieved from <https://subscription.packtpub.com/book/networking_and_servers/9781783550630/1/ch01lvl1sec09/the-ansible-architecture>
3. Ansible, R. H. (n.d.). AnsibleFest Atlanta - Network Automation of F5 BIG -IP Devices with Ansible Tower. Retrieved from <https://www.ansible.com/network-automation-of-f5-big-ip-devices-with-ansible-tower>
4. Ziouani, S. (n.d.). Enterprise companies welcoming Ansible IT Automation. Retrieved from <https://www.ansible.com/blog/enterprise-ansible>
5. London, D. (n.d.). New Case Study: NASA and Ansible Tower. Retrieved from <https://www.ansible.com/blog/nasa-automation>
6. Working With Playbooks. (2020, March 5). Retrieved from <https://docs.ansible.com/ansible/latest/user_guide/playbooks.html>
7. [IPvZero]. (2018, November 29). Ansible Playbooks for Cisco Network Automation! [Video File]. Retrieved from <https://www.youtube.com/watch?v=xw4IinFGVzA>
8. Ansible, Inc. (n.d.). Ansible Project. Retrieved from <https://docs.ansible.com/>
9. Ansible, R. H. (n.d.). Integrations Overview. Retrieved from <https://www.ansible.com/integrations>
10. Installing Ansible. (2020, March 5). Retrieved from https://docs.ansible.com/ansible/latest/installation\_guide/intro\_installation.html#prerequisites